



of a text area portion with color information. Upon decompression, the individual data are decompressed and combined to express an original image.

However, a compression ratio that can be realized  
5 by the aforementioned method while maintaining high image quality is not high enough. Also, information of characters emphasized in red in black character text is lost. That is, when an image containing a text portion using two or more colors is compressed by the above  
10 compression method and the compressed image is expanded, the number of colors of the text portion contained in the expanded image is reduced to one.

#### SUMMARY OF THE INVENTION

15 The present invention has been made to solve the conventional problems, and has as its object to provide an image processing apparatus and method which can efficiently compress an image while maintaining high image quality, and a storage medium for implementing  
20 the method.

According to present invention, the foregoing object is attained by providing an image processing apparatus comprising: extraction means for extracting  
a text area from multi-valued image data, and  
25 generating position data of the text area; color computation means for generating representative color data of a text portion in the text area; generation



compressing the text image data; and resolution  
conversion means for generating reduced non-text  
multi-valued image data by lowering a resolution of the  
non-text multi-valued image data, and wherein the  
5 first compression means compresses the reduced non-text  
multi-valued image data.

In still another aspect of the present invention,  
the foregoing object is attained by providing an image  
processing apparatus comprising: extraction means for  
10 extracting a text area from multi-valued image data,  
and generating position data of the text area; color  
computation means for generating representative color  
data of a text portion in the text area; generation  
means for generating text image data expressed by the  
15 representative color in the text area; conversion  
means for generating non-text multi-valued image data  
by converting multi-valued image data of the text  
portion using multi-valued image data of a portion  
other than the text portion; first compression means  
20 for compressing the non-text multi-valued image data;  
and second compression means for compressing the text  
image data wherein the color computation means  
includes: color palette generation means for  
generating at least one color palette as the  
25 representative color data by executing a color  
reduction process of a text image in the text area, and  
the generation means includes: color-reduced image

generation means for generating color-reduced image data corresponding to the color palette.

In still another aspect of the present invention, the foregoing object is attained by providingAn image processing apparatus comprising: extraction means for extracting a text area from multi-valued image data, and generating position data of the text area; color computation means for generating representative color data of a text portion in the text area; generation means for generating text image data expressed by the representative color in the text area; conversion means for generating non-text multi-valued image data by converting multi-valued image data of the text portion using multi-valued image data of a portion other than the text portion; first compression means for compressing the non-text multi-valued image data; and second compression means for compressing the text image data, wherein the generation means includes binarization means for applying a differential filter to the multi-valued image data, computing edge amounts of pixels that form the multi-valued image with neighboring pixels, and binarizing the data on the basis of the edge amounts.

In still another aspect of the present invention, the foregoing object is attained by providing an image processing apparatus comprising: extraction means for extracting a text area from multi-valued image data,

and generating position data of the text area; color computation means for generating representative color data of a text portion in the text area; generation means for generating text image data expressed by the representative color in the text area; conversion means for generating non-text multi-valued image data by converting multi-valued image data of the text portion using multi-valued image data of a portion other than the text portion; first compression means for compressing the non-text multi-valued image data; and second compression means for compressing the text image data, wherein the conversion means comprises: segmentation means for segmenting the multi-valued image into blocks each having a predetermined size; and discrimination means for discriminating if each of the blocks segmented by the segmentation means includes a text portion, and when the discrimination means discriminates that the block of interest includes a text portion, an average value of colors of a portion other than the text portion in the block of interest is computed, and the multi-value image data of the text portion is converted using the computed average color value.

Other features and advantages of the present  
25 invention will be apparent from the following  
description taken in conjunction with the accompanying









processing apparatus according to the third embodiment of the present invention;

Fig. 25 is a schematic block diagram of an image expansion apparatus corresponding to the image

5 processing apparatus according to the third embodiment of the present invention;

Fig. 26 is a flow chart of an image expansion process done by the image expansion apparatus according to the third embodiment of the present invention;

10 Fig. 27 is a schematic block diagram showing the arrangement of an image processing apparatus according to the fourth embodiment of the present invention;

Fig. 28 is a schematic block diagram showing the arrangement of an image processing apparatus according to the fifth embodiment of the present invention;

15 Fig. 29 is a block diagram showing the functional arrangement of the image processing apparatus according to the fifth embodiment of the present invention;

Figs. 30A to 30E are views for explaining an inclusion process of the image processing apparatus according to the fifth embodiment of the present invention;

Fig. 31 is a block diagram showing the functional arrangement of an image processing apparatus according to the sixth embodiment of the present invention;

Figs. 32A to 32F are views for explaining an inclusion process of the image processing apparatus





multi-valued image, the extracted text areas are  
 respectively binarized to obtain partial binary images  
 108, and the partial binary images 108 are compressed  
 by MMR to generate compressed codes 112. On the other  
 5 hand, the portions of the partial binary images 108 in  
 the entire multi-valued image are filled by  
 multi-valued images to generate image A, which is  
 reduced and compressed by JPEG, thus generating a  
 compressed code 111. Furthermore, the color  
 10 computation unit 103 obtains representative colors 110  
 of text portions for respective text areas.

<Text Area Detection Process>

Detailed contents of the text area detection  
 process will be explained below using Figs. 3 to 8.  
 15 Fig. 3 is a flow chart for explaining the processing in  
 the text area detector 101. Note that the program code  
 according to this flow chart is stored in a memory such  
 as a ROM, RAM, or the like (not shown) in the image  
 processing apparatus of this embodiment, and is read  
 20 out and executed by a CPU (not shown).

In step S301, an original image 100 is input, and  
 undergoes luminance conversion while being decimated to  
 lower its resolution, thus generating image J of  
 luminance Y. For example, if the original image 100 is  
 25 expressed by R, G, and B 24-bit data and has a  
 resolution of 300 dpi:

$$Y = 0.299R + 0.587G + 0.114B$$

is computed every four pixels in both the vertical and horizontal directions to generate new image J.

Luminance Y of new image J is expressed by 8 bits, and the resolution is 75 dpi. In step S302, the histogram

5 of the luminance data is computed to obtain

binarization threshold value T. In step S303, image J is binarized by threshold value T to generate binary image K. Furthermore, in step S304 the edges of black pixels are traced to label all black areas. In step

10 S305, a character candidate area in each black area is discriminated. In step S306, areas to be coupled are coupled based on their patterns and positions.

For example, when an original image shown in Fig. 4 is input, the histogram of this image after

15 decimation and luminance conversion is computed, as shown in Fig. 5. Threshold value  $T = 150$  is computed from this histogram using data such as an average,

variance, and the like, and an image binarized using this threshold value is as shown in Fig. 6. When the

20 edges of black pixels in Fig. 6 are traced to label all areas, and only sets of black pixels each having a width or height equal to or smaller than the threshold value are determined to be characters, the sets of black pixels shown in Fig. 7 form text areas (Fig. 7

25 shows a concept for the purpose of description, but such image is not generated in practice).

When these sets of black pixels are grouped based on near positions, and matches of widths and heights, 17 text areas can be detected, as shown in Fig. 8. The coordinate data of these areas are stored in a RAM (not shown) as text area coordinate data 109 shown in Fig. 1.

In place of binarizing the original image 100, the image 100 may be filtered by a differential filter to compute edge amounts of all pixels with neighboring pixels, and a binary image obtained by binarizing these edge amounts may undergo edge trace to detect text areas.

<Text Portion Compression Process>

In order to compress text portions, the binarization unit 102 generates partial binary images 108 of the 17 text areas obtained by the aforementioned method. Such binary image may be generated by binarizing the text area using T computed by the text area detector 101, or the histogram of each area may be computed to obtain an optimal binarization threshold value for that text area. Since the luminance histogram of each text area is expected to have a simple pattern shown in Fig. 9 compared to the histogram of the entire image shown in Fig. 5, it is easy to determine a threshold value. Reference numeral 901 denotes a set of background colors; and 902, a set of text colors.

09931890-082004  
T00280-0681660

The binary image data of the text portions obtained by binarizing the respective text areas are input to the MMR compression unit 107, are compressed by MMR, and are stored as a plurality of compressed codes 112 in a memory.

<Text Color Computation Process>

The text color computation process is done for each of all the extracted text areas. Fig. 12 shows an example of the text color computation unit using the binarization results.

It is checked in step S1201 if character area coordinate data to be processed still remain. If YES in step S1201, the flow advances to step S1202; otherwise, the flow advances to "end". In step S1202, a binary image in the text area addressed by that coordinate data undergoes thin-line conversion. That is, black pixels corresponding to change portions from the background to character portions upon scanning by a scanner are reduced to generate a new binary image "newbi". In step S1203, the R, G, and B histograms of the original image 100 corresponding to black pixels of "newbi" are computed (of course, other color spaces such as YUV and the like may be used). In step S1204, R, G, and B representative values are computed.

The method of computing the representative values includes the following representative value extraction method in addition to a method of simply selecting a



density having a maximum frequency of occurrence in the histogram as a representative value.

That is, in that method, a coarse histogram with a fewer number of steps is generated to obtain a  
5 density range including a maximum frequency of occurrence, and a density value having a maximum frequency of occurrence within that density range is selected as a representative value with reference to a fine histogram. This method will be explained in  
10 detail below with reference to the drawings.

Assume that the pixel densities of the original image 100 at a position corresponding to a binary image "newbi" in a given text area are expressed by a 256-step histogram, and an R histogram shown in Fig. 13  
15 is obtained. Simply calculated, the density value of the maximum frequency of occurrence in this histogram is 1302. However, most pixels which form this character gather around 1301, and fall within the range from 150 to 230 as the R density. Therefore, a value  
20 1301 is appropriate as a representative value.

In order to extract the value 1301 as the representative value, the total density range from 0 to 255 of the histogram is divided into nine overlapping ranges each having a 32-density width. In this example,  
25 the density range is divided into nine density ranges (0): 0 to 31, (1): 0 to 63, (2): 32 to 95, (3): 64 to 127, (4): 96 to 159, (5): 128 to 191, (6): 160 to 223,

(7): 192 to 255, and (8): 224 to 255. Although each of ranges (1) to (7) has a 64-density width, each of ranges (0) and (8) has a 32-density width, and the number of pixels belonging to these ranges is doubled  
5 upon comparison.

When a coarse histogram is generated by counting the numbers of pixels belonging to these density ranges, the number of pixels that belong to range (6) is largest in Fig. 13. Hence, the maximum value of range  
10 (6) is searched using a fine histogram to obtain 1301. The aforementioned process is repeated for all text areas, thus computing one representative color for each of all text areas.

<Background Image Compression Process>

15 Upon compressing a background image, a text portion paint process is done to extract an image that does not contain any text from the original image 100.

An example of the processing of the text paint unit 104 using the binarization results will be  
20 explained below using Figs. 10A to 10C and Fig. 11. Figs. 10A to 10C show an example of images to explain the text portion paint process, and Fig. 11 is a flow chart showing the flow of the text portion paint process. The program code according to the flow chart  
25 shown in Fig. 11 is stored in a memory such as a ROM, RAM, or the like (not shown) in the image processing

apparatus of this embodiment, and is read out and executed by a CPU (not shown).

Assume that the original image 100 of a text area is an image which has a gradation image as a background and on which blue letters "ABC" are drawn near the center, as shown in Fig. 10A. Also, a binary image shown in Fig. 10B is obtained from this original image 100. In this embodiment, the image is broken up into  $32 \times 32$  areas (to be referred to as parts hereinafter), and processing is done for respective parts. Fig. 10C shows the image broken up into parts. Fig. 10C illustrates a state wherein the image is broken up into  $5 \times 4$  parts for the sake of simplicity. The upper left numeral of each part indicates a part number.

When the image is broken up in this way, since it is determined in step S1103 that parts 00 to 04, 10, 14, 20, 24, and 30 to 35 do not contain any binary image, the control advances to the processing for the next part without any processing. For part 11, the flow advances to step S1104 to compute an average value ave\_color of R, G, and B values (or may be Y, U, and V values or the like) of the original image 100 corresponding to a white portion of the binary image. In step S1105, the corresponding binary image is looked up, and ave\_color is substituted in density data of pixels corresponding to black pixels. That is, the density of the text image is converted into the average

09931890.082001  
T00280.058T560

density of the image around text. The aforementioned process is repeated for parts 12, 13, 21, 22, and 23 containing the text area.

5 In this way, the text portion is filled with the average value of pixel densities around that portion. Then, the image in which the text portion is painted is reduced by the reduction unit 105, and is compressed by JPEG to generate a compressed code 112. In this embodiment, reduction in the reduction unit 105 is  
10 implemented by simple decimation. Note that the order of this reduction process and text portion paint process may be reversed. In this case, a positional deviation, if any, between the binary image and reduced image must be taken care of.

15 <Format Generation>

A format that combines four data, i.e., the text area coordinate data 109, text portion representative colors 110, and compressed codes 111 and 112 may be generated as needed. This format may be the PDF or XML.

20 Alternatively, these four data, i.e., the text area coordinate data 109, text portion representative colors 110, and compressed codes 111 and 112 may be compressed by a reversible compression method together.  
<Expansion Process>

25 The arrangement for explaining an image compressed by the image processing apparatus of this embodiment will be explained below using Fig. 2.

Reference numeral 201 denotes a JPEG expansion unit for receiving the compressed code 111, and expanding it by JPEG to generate multi-valued image E. Reference numeral 202 denotes an MMR expansion unit for receiving the compressed code 112 to generate a binary image 205. Reference numeral 203 denotes an enlargement unit for receiving the multi-valued image and enlarging it to generate an image 206. Reference numeral 204 denotes an image combining unit for receiving the coordinate and representative color data of text areas, selecting the image 206 for a white portion and the corresponding representative color for a black portion with reference to the binary image 205, and generating an expanded image 207 as a final image.

Figs. 14A to 14C show an example of the result of the combining unit 204. Fig. 14A shows the JPEG expansion result of the compressed code 111. When a quantization irreversible scheme of JPEG is used, data having slightly different pixel values from Fig. 10C is obtained. However, pixel values change less than a case wherein the original image 100 before the text portion is extracted is compressed by the JPEG irreversible compression scheme, if the same quantization table is used. The representative color data (20, 30, 255) are superposed on corresponding image portions of black pixels with reference to the binary image expanded, as shown in Fig. 14B, thus



For example, only the text area coordinate data and original image 100 may be input, and a new binarization result of the original image 100 may be used in the color computation process. In such case, the text  
5 paint unit 104 and color computation unit 103 may respectively incorporate optimal binarization means in the arrangement shown in Fig. 16.

In this embodiment, the text area detector finally couples areas having similar patterns and  
10 positions in step S306 in Fig. 3, but they need not be coupled. In this case, 100 or more text areas are extracted from the image shown in Fig. 4, but the subsequent color computation process can be flexibly done. For example, even when most of 20 characters in  
15 a line are black but include some red characters, such change in color can be flexibly coped with.

In this embodiment, a high-luminance character (reversed character) on a low-luminance background cannot be processed. However, this process is enabled  
20 when, for example, the arrangement shown in Fig. 17 is adopted. Reference numeral 1701 denotes a differential processor, which applies a differential filter shown in Fig. 18A or 18B to pixels including the pixel of interest as the center, and binarizes a pixel to black  
25 if the absolute value exceeds a threshold value or white if the absolute value does not exceed the threshold value. Fig. 18A shows first-order











of the color reduction unit 1082 in the text color  
extraction unit 2108. The MMR compression unit 107  
receives color-reduced images 2109 when each  
color-reduced image 2109 consists of 1 bit (binary),  
5 and compresses them by MMR to generate a plurality of  
compressed codes 2115 corresponding to these  
color-reduced images 2109. Reference numeral 2111  
denotes a reversible compression unit for receiving  
color-reduced images 2109 when each color-reduced image  
10 2109 consists of 2 bits or more, and reversibly  
compressing them (e.g., by ZIP) to generate a plurality  
of compressed codes 2116 corresponding to these  
color-reduced images 2109.

The text area coordinate data 109, compressed  
15 code 111, palettes 2114, compressed codes 2115, and  
compressed codes 2116 are combined to obtain compressed  
data. If all text areas are expressed by 1 bit, no  
compressed codes 2116 are generated. Also, if no text  
areas are present, compressed data 1A contains only  
20 compressed code 111.

Fig. 22 is a flow chart showing the process in  
the text color extraction unit 2108 including the color  
reduction unit 1082.

In step S2201, a counter num indicating the  
25 number of extracted colors is reset to zero, since  
processing is done for each of all extracted text areas.

It is checked in step S2202 if text coordinate data to be processed still remain. If YES in step S2202, the flow advances to step S2203; otherwise, this processing ends.

5           In step S2203, that portion of a binary image 2103, which is located at the text coordinate position to be processed, undergoes thin-line conversion, so as to reduce black pixels corresponding to change portions from the background to character portions upon scanning  
10 by a scanner, thereby generating a new binary image "newbi".

          In step S2204, an RGB three-dimensional histogram of the original image corresponding to black pixels of the image "newbi" is computed. When histograms are  
15 normally computed, if an input image is expressed by 8 bits for each of R, G, and B pixels,  $256 \times 256 \times 256$  histograms are required. Since a text portion requires high resolution but does not require excellent tone reproduction, and some pixel value differences are  
20 preferably ignored upon computing a representative color while suppressing variations at the time of scanning by a scanner, such fine histograms are not required. Hence, in this embodiment, a three-dimensional histogram of upper 5 bits of R, G,  
25 and B 8-bit data is computed. Upon computing the histogram, the total number pixelnum of black pixels present in that text area is also computed.





black pixel of the binary image within the text area on the basis of the original image 100 and binary image 103. As the number of bits to be assigned, since data indicating transparency is required as a white portion of the binary image, when the number of palettes is 3, 2 bits are assigned since three plus one for transparency data equals four. On the other hand, when the number of palettes is 4, 3 bits are assigned since four plus one for transparency data equals five. The number of bits is determined according to that a finally applied image format can express.

In this case, when the number of palettes generated as a result of the color reduction process of a given text area is considerably large, and it is determined that effects of improvements of image quality and compression ratio by color reduction cannot be expected, that text area may be saved as a background image. In this case, it must be determined in the process of the text paint unit 104 that the text area is not processed as a text area and does not undergo a text portion paint process.

Fig. 24 is a flow chart for generating compressed data 1A. Note that the program code according to this flow chart is stored in a memory such as a ROM, RAM, or the like (not shown) in the image processing apparatus of this embodiment, and is read out and executed by a CPU (not shown).





A schematic arrangement of an image expansion apparatus for expanding the compressed data 1A compressed as described above will be explained below using Fig. 25. Fig. 26 is a flow chart of an image expansion process done by the image expansion apparatus of this embodiment. Note that the program code according to this flow chart is stored in a memory such as a ROM, RAM, or the like (not shown) in the image processing apparatus of this embodiment, and is read out and executed by a CPU (not shown).

Reference numeral 201 denotes a JPEG expansion unit for receiving the compressed code 111, and generating a multi-valued image by executing a JPEG expansion process (step S2601). Reference numeral 203 denotes an enlargement unit for receiving the multi-valued image, and generating an image 206 by executing an enlargement process (step S2602).

Reference numeral 202 denotes an MMR expansion unit for receiving the compressed code 2115 and generating a binary image 205 by executing an MMR expansion process (step S2603). Reference numeral 2506 denotes a reversible expansion unit for receiving the compressed code 2116 and generating a multi-color image 2507 (step S2604). Reference numeral 2508 denotes a combining unit for receiving the text area coordinate data 109, corresponding palettes 2114, and binary image 205 or multi-color image 2507, and generating an image

2509 as a final image by selecting the pixel color of the image 206 when pixel data of the binary image or multi-color image indicates transparency, and selecting a corresponding pallet color in other cases.

5           On the other hand, when a text area is a  
multi-color image, the number of palettes changes (step  
S2605). For example, if 2 bits are assigned, palettes  
assigned to four pixel values 00, 01, 10, and 11 are  
selected and applied (step S2606). One (e.g., 00) of  
10 these pixel values indicates transparency, and the  
pixel density of the JPEG-expanded image 206 is  
selected for a pixel position having the value 00.

The aforementioned image expansion apparatus and method expand the compressed data of this embodiment to  
15 reclaim an image, and can assign a plurality of colors to a given text area when the text area contained in that image originally has a plurality of colors.

In the above embodiment, a background image undergoes JPEG compression, a text binary image undergoes MMR compression, and a text multi-valued image undergoes ZIP compression. However, the present invention is not limited to this, and three types of compression methods, i.e., a first compression method suitable for a 256-level multi-valued image, a second compression method suitable for a binary image, and a third compression method suitable for a multi-valued image having around 16 levels need only be used. The

second and third compression methods may be the same method.

(Fourth Embodiment)

In the third embodiment, the text area detector  
5 101 binarizes a color image. Alternatively, a  
differential filter may be applied to an original image  
to compute edge amounts of all pixels with neighboring  
pixels, and a binary image obtained by binarizing these  
edge amounts may undergo edge trace to detect text  
10 areas. Fig. 27 shows a schematic arrangement of an  
image processing apparatus in this case. Since the  
respective building components are the same as those  
described in Figs. 17 and 21, the same reference  
numerals denote the same building components and a  
15 detailed description thereof will be omitted.

In this embodiment, since the text paint unit 104  
and text color extraction unit 2108 cannot use a binary  
image, a binary image for each text area is generated.

In this way, high-luminance characters (reversed  
20 characters) on a low-luminance background can be  
processed.

(Fifth Embodiment)

As the fifth embodiment, an image processing  
apparatus which compresses more efficiently using the  
25 image processing method described in the first  
embodiment will be described below. This embodiment  
couples text areas, to which an identical

representative color is assigned by the image process described in the first embodiment, to handle them as a single text area.

Fig. 28 shows the basic arrangement of an image processing apparatus of this embodiment. A CPU 2801 controls the overall apparatus of this embodiment and executes an image compression process using programs and data stored in a RAM 2802 and ROM 2803. The RAM 2802 has an area for temporarily storing programs and data loaded from an external storage device 2804 and storage medium drive 2809, image data input from an image input device 2808, and the like, and also a work area used by the CPU 2801 upon executing various processes. The ROM 2803 stores control programs and boot program of the overall apparatus, setup data of the apparatus, and the like. The external storage device 2804 comprises, e.g., a hard disk, and can save programs, data, and the like loaded from the storage medium drive 2809. Also, when the work area size has exceeded the size of the RAM 2802, the device 2804 can provide the extra area as a file. A keyboard 2805 and mouse 2806 serve as pointing devices, which allow the user to input various instructions to the apparatus of this embodiment.

A display device 2807 comprises a CRT, liquid crystal display, or the like, and can display image information and text information. The image input

device 2808 comprises a scanner, digital camera, or the like, can input an image as data, and includes an interface for connecting the apparatus of this embodiment. The storage medium drive 2809 comprises a CD-ROM drive, DVD-ROM drive, floppy disk (FD) drive, or the like, and can load programs, data, and the like from a storage medium such as a CD-ROM, DVD-ROM, FD, or the like. Reference numeral 2810 denotes a bus for connecting the aforementioned units.

Fig. 29 shows the functional arrangement of the image processing apparatus of this embodiment. Characters contained in a color document image 2901 use a plurality of colors. A binarization unit 2902 binarizes the color document image 2901 to generate a binary image. An area analysis unit 2903 corresponds to the text area detector 101 in Fig. 1, specifies text areas in the binary image, and generates information (text image information) including the positions, sizes, and the like of the text areas in the binary image. As a method of specifying text areas, for example, a method of tracing the edges of black pixels in the binary image, labeling all black areas, and specifying a character candidate area (i.e., an area which is likely to be a text area) in each black area by searching for labeled black areas may be used. However, the present invention is not limited to such specific method.

20

25



Fig. 28 via the storage medium drive 2809. In this case, when the CPU 2801 executes the loaded program, the image processing apparatus with the arrangement shown in Fig. 28 serves as the apparatus having the functional arrangement shown in Fig. 29.

A color document image compression method in this embodiment will be described below using the functional block diagram of Fig. 29.

A color document image 2901 is loaded from the external storage device 2804, image input device 2808, or storage medium drive 2809 onto the RAM 2802. In this embodiment, an image shown in Fig. 30A is used as the color document image 2901.

The binarization unit 2902 generates a binary image on the basis of the color document image 2901 loaded onto the RAM 2802. A method of generating the binary image is not particularly limited, and this embodiment uses the following method. A histogram of luminance data in the color document image 2901 is computed to obtain a binarization threshold value T. This computation method is not particularly limited and, for example, a luminance value as an intermediate value of the histogram may be selected as the threshold value T. The color document image 2901 is binarized using the binarization threshold value T to generate a binary image. The generated binary image is stored in an area



different from that which stores the color document  
image 2901 in the RAM 2802.

The area analysis unit 2903 specifies text areas with reference to the binary image generated by the binarization unit 2902. In this case, the  
5      aforementioned text image information is generated. If the image shown in Fig. 30A is to undergo area analysis of the area analysis unit 2903, the result shown in Fig. 30B is obtained. In Fig. 30B, text areas TEXT1 to  
10     TEXT5 are specified, and text image information is generated for each area. The text image information is output to the color extraction unit 2904 and text paint unit 2907.

The color extraction unit 2904 specifies text areas in the color document image 2901 with reference to the text image information, and extracts colors in the specified text areas, i.e., text colors in the text areas. In this embodiment, the areas TEXT1 and TEXT3 contain red characters, the areas TEXT2 and TEXT4 contain black characters, and the area TEXT5 contains blue characters in Fig. 30B. The colors of the respective text areas extracted by the color extraction unit 2904 are generated as color information.

The image coupling unit 2905 couples text areas  
25 using identical colors with reference to the color  
information of the text areas TEXT1 to TEXT5, which are  
extracted by the color extraction unit 2904. In this

embodiment, only text areas from which one representative color is extracted are coupled.

In the example shown in Fig. 30B, since the areas TEXT1 and TEXT3 contain characters using the same color, an image (inclusive image) of an area including these areas is generated. Note that generating an image including TEXT1 and TEXT3 is described as "coupling TEXT1 and TEXT3". In Fig. 30C, this inclusive image is indicated by TEXT1'. Note that pixels other than the text portion in this inclusive image are set to have a monochrome pixel value (e.g., density 0). The same applies to TEXT2 and TEXT4. Note that an inclusive image that includes TEXT2 and TEXT4 is indicated by TEXT2' in Fig. 30C.

Fig. 30D shows details of inclusive images TEXT1' and TEXT2'. The image coupling unit 2905 generates inclusive image information that contains the positions and sizes of the respective inclusive images (in the binary image or color document image 2901).

The method of specifying text areas using an identical color in the image coupling unit 2905 will be explained below. When the color extraction unit 2904 extracts a color expressed by 8 bits for each of R, G, and B, a color reduction process is done to obtain a predetermined color range (e.g., 2 or 3 bits for each of R, G, and B). The color obtained as a result of the color reduction process is redefined as a

representative color, and areas having an identical color are coupled. The color reduction limit is determined by desired tone reproduction of an image.

5   expressed by 2 bits, 2 bits, and 1 bit or 3 bits, 3  
bits, and 2 bits by exploiting the fact that the human  
eye is insensitive to blue.

When an identical color is to be discriminated more accurately, color data may be converted from an RGB format into an LAB or YCrCb format that can compare color differences more easily, and the converted color data may be rounded to 2 or 3 bits upon discrimination. In the RGB format, when black is compared with gray and dark blue, dark blue is closer to black. However, in the LAB or YCrCb format, since the luminance and color components are separated, black and dark blue can be separated.



areas, while in this embodiment, three headers (headers of TEXT1', TEXT2', and TEXT5) are generated. As a result, the number of headers can be decreased, and the compressed data size can be reduced.

5           The text paint unit 2907 specifies text areas in the color document image 2901 using the text image information, and generates an image (background image) obtained by painting the specified text areas by a given color. Fig. 30E shows this background image.

10       The given color may be a predetermined color or may be the average value of pixels around the text area in the color document image 2901.

          The background image compression unit 2908 compresses the image (background image) generated by  
15       the text paint unit 2907.

          As described above, according to the image processing apparatus and method of this embodiment, even when a color document image including many text areas is compressed, since an image including text  
20       areas having an identical color is generated and compressed, the number of headers to be appended to the compressed image can be reduced. At the same time, the compressed data size can be reduced.

(Sixth Embodiment)

25           As the sixth embodiment, an image processing apparatus which compresses more efficiently using the image processing method described in the second

embodiment will be described below. In this embodiment,  
one text area to which a plurality of color palettes  
are assigned by the image processing method described  
in the second embodiment is segmented into sub-areas,  
5 and areas are then coupled as in the fifth embodiment.

Fig. 31 shows the functional arrangement of an  
image processing apparatus of this embodiment. A color  
extraction unit 2904 corresponds to a text color  
extraction unit 2108 in Fig. 21, and generates a  
10 color-reduced image and color palettes by executing a  
color reduction process of a text image. An image  
coupling unit 2905 includes a text area segmentation  
unit 2905a. The text area segmentation unit 2905a  
segments a text area including a plurality of colors  
15 into sub-areas of identical colors. Since other  
arrangements are the same as those in Fig. 29, the same  
reference numerals denote the same parts, and a  
detailed description thereof will be omitted. In this  
embodiment, all text areas including a plurality of  
20 colors are segmented into sub-areas containing text of  
identical colors to obtain new text areas.

If an image shown in Fig. 32A is to undergo area  
analysis of the area analysis unit 2903, the area  
analysis result is as shown in Fig. 32B. The color  
25 extraction unit 2904 specifies text areas in the color  
document image 2901 with reference to the text image  
information, and extracts colors in the specified text

areas, i.e., text colors in the text areas. In this embodiment, the areas TEXT1 and TEXT3 contain red characters, the area TEXT2 contains black, red, and blue characters, the area TEXT4 contains black

5 characters, and the area TEXT5 contains blue characters  
in Fig. 32B. The color extraction unit 2904 extracts  
the colors of the respective text areas as color  
palette information.

The image coupling unit 2905 couples text areas  
10 using identical colors with reference to the color  
palette information of the text areas TEXT1 to TEXT5.

Since the area TEXT2 has three colors, the text area segmentation unit 2905a segments the area TEXT2 into areas (sub-areas) including identical color characters, as shown in Fig. 32C. An area including black characters in TEXT2 is indicated by TEXT6. An area including red characters in TEXT2 is indicated by TEXT7. An area including blue characters in TEXT2 is indicated by TEXT8. The palette information of TEXT2 is also segmented to generate black color palette information, red color palette information, and blue color palette information in correspondence with the sub-areas TEXT6, TEXT7, and TEXT8. Also, the text area segmentation unit 2905a generates sub-area information containing the positions and sizes of these sub-areas.

As the method of segmenting a text area having a plurality of colors into sub-areas, the text area

further undergoes a color reduction process, and is segmented into sub-areas including areas having identical colors using the color reduction result.

Upon completion of the sub-area segmentation  
5 process by the text area segmentation unit 2905a, text  
areas (including sub-areas) using identical colors are  
coupled to generate inclusive images.

Since the areas TEXT1, TEXT3, and TEXT7 include characters using an identical color (red), an inclusive image including these areas is generated. This inclusive image area is indicated by TEXT1' in Fig. 32D.

Likewise, an area TEXT2' including areas having black characters is generated. TEXT2' includes TEXT4 and TEXT6. Also, an area TEXT3' including text areas 15 having blue characters is generated. TEXT3' includes TEXT5 and TEXT8.

Fig. 32E shows the contents of the inclusive images TEXT1', TEXT2', and TEXT3'. Such inclusive images are output to the binary image compression unit 2906. The image coupling unit 2905 generates inclusive image information containing the positions and size (in the binary image or color document image 2901) of the respective inclusive images.

The image coupling unit 2905 directly outputs  
25 images of a text area and sub-area which are not  
coupled to the binary image compression unit 2906.



The binary image compression unit 2906 compresses the respective inclusive images and/or text areas and/or sub-areas. Since all the inclusive areas, text areas, and sub-areas consist of 1-bit images (having  
5 one color), MMR compression is used. As a result, compression efficiency can become higher than reversible compression (ZIP compression) used upon compressing a text area having a plurality of colors.

Upon compressing an inclusive image, a header  
10 corresponding to that inclusive image is generated, and contains color information and inclusive image information of that inclusive image. Upon compressing a text area, a header corresponding to this text area is generated, and contains color information and text  
15 image information of the text area. Upon compressing a sub-area, a header corresponding to this sub-area is generated, and contains color information and sub-area information of the sub-area.

As the color information of an inclusive image,  
20 color palette information of one of text areas (including sub-areas) in the inclusive image is used.

Upon independently compressing text areas (including sub-areas), seven headers (those of TEXT1, TEXT3, TEXT4, TEXT5, TEXT6, TEXT7, and TEXT8) are  
25 generated, while in this embodiment, three headers (those of TEXT1', TEXT2', and TEXT3') are generated. As a result, the number of headers can be decreased,

and the compressed data size can be reduced.

The text paint unit 2907 specifies text areas in the color document image 2901 using the text image information, and generates an image (background image: 5 Fig. 32F) obtained by painting the specified text areas by a given color.

As described above, according to the image processing apparatus and method of this embodiment, even when a color document image including many text areas is compressed, since an image including text areas having an identical color is generated and compressed, the number of headers to be appended to the compressed image can be reduced. At the same time, the compressed data size can be reduced.

15           Note that a text area having a plurality of  
colors may be broken up into sub-areas, which may be  
independently compressed without being coupled. That  
is, MMR compression of individual segmented sub-areas  
often assures higher compression efficiency than  
20 reversible compression of a text area assigned a  
plurality of colors even when they are not coupled.  
(Seventh Embodiment)

In the fifth and sixth embodiments, text areas having an identical color are included in a single inclusive image, which is compressed by MMR. However, when separate small text areas having an identical color are included in the inclusive image, the size

after compression may often increase. In this embodiment, it is checked if text areas having an identical color are to be coupled to generate an inclusive image, thus realizing more effective

5 compression.

More specifically, the image coupling unit 2905 in Fig. 29 executes an inclusion discrimination process in addition to the processes described in the fifth and sixth embodiments. Other building components are the same as those in the fifth and sixth embodiments.

The processing of the image coupling unit 2905 of this embodiment will be explained below using Fig. 33. Fig. 33 is a flow chart of the detailed processing in the image coupling unit 2905 of this embodiment.

15           The image coupling unit 2905 selects one text  
area serving as a reference (to be referred to as a  
reference text area hereinafter) from text areas which  
are determined to have an identical color (step S3301).  
If no text area is available or if the processing is  
20   complete for all text areas (step S3302), this  
processing ends. On the other hand, if text areas to  
be processed still remain, the flow advances to step  
S3303.

Text areas which neighbor the reference text area  
25 and have an identical color are searched for (step  
S3303), and if areas which meet this condition are  
found, the flow advances to step S3304 to select a text

area (to be referred to as a neighboring text area hereinafter) which meets the condition and is closest to the reference text area (step S3304). On the other hand, if a text area that meets the condition is not  
5 found, the flow advances to step S3309 to generate an inclusive image which includes text areas which are determined in step S3308 (to be described later) as those to be coupled to the reference text area (step S3309).

10           An inclusive image rectangle that includes the reference text area and neighboring text area is determined (step S3305). The total size of compressed data upon individually compressing the reference text area and neighboring text area, and the compressed size  
15 upon compressing an inclusive image are estimated (step S3306). When two areas (reference text area and neighboring text area) are individually compressed using compression ratio A of a text area which is measured in advance, their total size can be estimated  
20 by:

$$\begin{aligned} \text{Compressed size 1} = & (\text{area of reference text area} \\ & + \text{area of neighboring text area}) \times A \\ & + 2 \times \text{header size} \end{aligned}$$

On the other hand, when an inclusive image is  
25 compressed, the two areas, i.e., the reference text area and neighboring text area included in the inclusive image always has a gap. This gap is filled

with data representing a given pixel value, and can be compressed at a greatly higher compression ratio than that upon compressing a text area. If B represents this compression ratio, the size of the compressed

5 inclusive image is given by:

$$\text{Compressed size 2} = (\text{area of text area}) \times A + (\text{area of gap}) \times B + \text{header size}$$

In step S3306, images may be actually compressed to obtain accurate sizes. However, when the sizes are  
10 calculated by a simple method, the processing time can be shortened.

Compressed sizes 1 and 2 are compared using the estimation results (step S3307). If it is determined that compressed size 2 is smaller than compressed size 1, i.e., if the compressed inclusive image has a smaller compressed data size than that obtained by individually compressing areas, the flow advances to step S3308 to add data indicating that the reference text area and neighboring text area are included (coupled) in a single inclusive image to a coupling list (step S3308).

Fig. 34 shows an example of the coupling list. Fig. 34 shows a configuration example of the coupling list when TEXT2 is a reference text area, and shows correspondence between TEXT2 and respective text areas TEXT1 to TEXT5. In Fig. 34, "0" is a non-coupled code, "1" is a coupled code, and "999" is an invalid code

(TEXT2 cannot be coupled to itself). Initially,  
non-coupled codes (0 in Fig. 34) are set in all fields  
of the coupling list, and are changed to a coupled code  
(1 in Fig. 34) only when the process in step S3308 is  
5 executed.

If it is determined in step S3307 that compressed  
size 2 is larger than compressed size 1, i.e., if the  
compressed inclusive image has a larger compressed data  
size than that obtained by individually compressing  
10 areas, the flow returns to step S3303 to search for the  
next neighboring text area.

After the first loop of the aforementioned  
processes, when the reference text area and neighboring  
text area are coupled, a text area which is other than  
15 the text area selected once, has the same color as the  
reference text area, and is closest to the reference  
text area, is selected as a new neighboring text area  
in the next loop of the processes in step S3303 and  
subsequent steps (steps S3303 and S3304). An inclusive  
20 image (second inclusive image) rectangle that includes  
the reference text area, previous neighboring text area,  
and current neighboring text area is determined (step  
S3305), and compressed sizes 1 and 2 of the second  
inclusive image and current neighboring text area are  
25 estimated using the above equations (step S3306). More  
specifically, the following equations are used:

Compressed size 1 = (area of second inclusive  
image + area of neighboring text  
area)  $\times$  A + 2  $\times$  header size

Compressed size 2 = (area of text area)  $\times$  A +  
5 (area of gap)  $\times$  B + header size

Then, the processes in step S3307 and subsequent  
steps are executed. In this manner, an inclusive image  
which includes a largest number of text areas and has  
the smallest compressed size can be generated.

10 (Another Embodiment)

Note that the present invention may be applied to  
either a part of a system constituted by a plurality of  
devices (e.g., a host computer, interface device,  
reader, printer, and the like), or a part of an  
15 apparatus consisting of a single equipment (e.g., a  
copying machine, facsimile apparatus, or the like).

The present invention is not limited to the  
apparatus and method alone for implementing the  
aforementioned embodiments, but the scope of the  
20 present invention includes a case wherein the above  
embodiments are achieved by supplying a program code of  
software that can implement the functions of the  
above-mentioned embodiments to a computer (or a CPU or  
MPU) in the system or apparatus, and making the  
25 computer control various devices in the system or  
apparatus.

In this case, the program code itself of the software implements the functions of the above-mentioned embodiments, and the program code itself, and means for supplying the program code to the computer (i.e., a storage medium which stores the program code) are included in the scope of the present invention.

As the storage medium for storing such program code, for example, a floppy disk, hard disk, optical disk, magneto-optical disk, CD-ROM, DVD, magnetic tape, nonvolatile memory card, ROM, and the like may be used.

The program code is included in the scope of the embodiments not only when the functions of the above embodiments are implemented by controlling various devices according to the supplied program code alone but also when the functions of the embodiments are implemented by collaboration of the program code and an OS (operating system) or another application software running on the computer. Furthermore, the scope of the present invention includes a case wherein the functions of the above-mentioned embodiments are implemented by some or all of actual processing operations executed by a CPU or the like arranged in a function extension board or a function extension unit, which is inserted in or connected to the computer, after the supplied program code is written in a memory of the extension board or unit.



To restate, according to the above embodiments,  
an image processing apparatus and method for  
efficiently compressing an image while maintaining high  
image quality, and a storage medium for implementing  
5 that method can be provided.

As many apparently widely different embodiments  
of the present invention can be made without departing  
from the spirit and scope thereof, it is to be  
understood that the invention is not limited to the  
10 specific embodiments thereof except as defined in the  
appended claims.

09931890-032004  
F00280-061866